

Nama : RIZQILLAH

NIM : 1957301020

MK : Prak. Pemrograman Lanjut

PEMROGRAMAN BERORIENTASI OBJEK

1.1 Object Oriented Programming

OOP (Object Oriented Programming) adalah suatu metode pemrograman yang berorientasi kepada objek. Tujuan dari OOP diciptakan adalah untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada di kehidupan sehari-hari. Setiap bagian dari suatu permasalahan adalah objek, objek itu sendiri merupakan gabungan dari beberapa objek yang lebih kecil lagi. Saya ambil contoh Sepeda Motor, pada sepeda motor pasti memiliki bagian-bagian dari yang terbesar sampai bagian terkecil. Dan bagian-bagian inilah yang disebut objek-objek yang memiliki class-class nya masing-masing. Dan setiap class-class tersebut bisa saja memiliki method yang banyak ataupun method yang sedikit. Setiap method itu bisa kita gunakan agar melakukan pekerjaan yang diinginkan, yaitu seperti jika kita membuat method pada lampu sepeda motor agar hidup, maka jika kita memanggil method tersebut, maka lampu sepeda motor tersebut akan hidup.

Begitu juga dengan program, sebuah objek yang besar dibentuk dari beberapa objek yang lebih kecil, objek-objek itu saling berkomunikasi, dan saling berkiriman pesan kepada objek yang lain.

1.2 Program Pertambahan

Coding :

```
package Tambah;
```

```
class aritmatika{
    private int a;
    private int b;

    public aritmatika(int a, int b){
        this.a = a;
        this.b = b;
    }

    public void setTambah(int a, int b){
```

```

        this.a = a;
        this.b = b;
    }

    public int getTambah(){
        return this.a + this.b;
    }

    public void show(){
        System.out.println("Hasil dari " + this.a + " + " +
this.b + " adalah " + this.getTambah());
    }
}

public class Main {
    public static void main(String[]args){
        aritmatika tambah = new aritmatika(5,5);
        tambah.show();

        tambah.setTambah(100, 50);
        tambah.show();
    }
}

```

Hasil :

```

Hasil dari 5 + 5 adalah 10
Hasil dari 100 + 50 adalah 150
BUILD SUCCESSFUL (total time: 0 seconds)

```

Analisa :

package Tambah;

➔ Membuat Nama Package(folder) tempat Program Berada

class aritmatika{

➔ Membuat Class dengan nama aritmatika

private int a;

private int b;

➔ Membuat variable(atribut) dengan access private

```

public aritmatika(int a, int b){
    ➔ Membuat Construct yang menerima inputan parameter langsung
      dari pembuatan objek

        this.a = a;
        this.b = b;
    ➔ Membuat nilai yang diinputan pada parameter construct agar
      masuk kedalam variable global pada class

}

public void setTambah(int a, int b){
    ➔ Membuat Methods yang menerima inputan parameter

        this.a = a;
        this.b = b;
    ➔ Membuat variable global menerima hasil inputan dari
      parameter Methods

}

public int getTambah(){
    ➔ Membuat Mathods untuk melakukan pertambahan terhadap 2
      variable

        return this.a + this.b;
    ➔ Mengembalikan hasil pertambahan 2 variable

}

public void show(){
    ➔ Membuat method yang berfungsi menampilkan hasil ke layer

        System.out.println("Hasil dari " + this.a + " + " +
this.b + " adalah " + this.getTambah());
    ➔ Menampilkan hasil pertambahan yang dilakukan

}
}

public class Main {
    ➔ Membuat class public dari file java

```

```

public static void main(String[]args){
→ Membuat fungsi main dari program

    aritmatika tambah = new aritmatika(5,5);
→ Membuat objek dari class aritmatika sambil mengisi nilai
    parameter yang akan diterima oleh construct

    tambah.show();
→ Memanggil Methods show() yang akan menampilkan hasil ke
    layar

    tambah.setTambah(100, 50);
→ Melakukan perubahan terhadap nilai variable dengan
    menggunakan methods

    tambah.show();
→ Memanggil methods show() untuk menampilkan hasil terkini
    dari variable

    }
}

```

KESIMPULAN

Dapat kita simpulkan, bahwa dalam membuat program, OOP(Object Oriented Programming) sangat dibutuhkan, agar program yang kita buat bisa menjadi lebih simple, elegan, mudah dimodifikasi, dan lain-lain.

Untuk membuat program agar bisa dikatakan telah memiliki OOP adalah dengan adanya beberapa dari konsep Class, Object, attribute, Method, Construct, Inheritance, Polymorphism, Encapsulation, dan Abstract.